A Rescoring Method Using Web Search and Word Vectors for Spoken Term Detection

Haruka Tanji* , Kazunori Kojima* , Hiroaki Nanjo† , Shi-wook Lee ‡ , and Yoshiaki Itoh*
*Iwate Prefectural University, Japan, E-mail:y-itoh@iwate-pu.ac.jp
†Academic Center for Computing and Media Studies, Kyoto University, Japan, E-mail:nanjo@media.kyoto-u.ac.jp
‡ National Institute of Advanced Industrial Science and Technology, Japan, E-mail:s.lee@aist.go.jp

ABSTRUCT

We propose a rescoring method using words related to a query obtained by Web search and word vectors for spoken term detection (STD). In this paper, we assume that words associated with the topic in speech data and co-occurring with the query are called “words related to the query”, and that the related words appear multiple times in the speech data. To identify the words related to the query, we introduce distributed expression of words obtained by Word2vec [1][2], and first convert each word in the word-recognition results of speech data into a word vector. Each word vector is then compared with a word vector of the query. Words related to the query are determined by calculating the degree of similarity between the two word vectors. However, a word vector of an out-of-vocabulary (OOV) query cannot be obtained in this manner, since OOV queries do not appear in word-recognition results. For such OOV queries, we perform a Web search using the query, whereupon texts including the query are extracted. Recognition results of the speech data and the extracted texts are then combined and used for training of Word2vec. In this manner, a word vector of the OOV query can be obtained. Distances to all candidates in the document, including words related to the query, are used advantageously. Experiments are conducted to evaluate the performance of the proposed method using open test collections of the NTCIR-10[3] and NTCIR-12[4] workshops. For retrieval accuracy, an improvement of 3.2 points in mean average precision was achieved using the proposed method.

I. INTRODUCTION

Spoken-document retrieval (SDR) and spoken-term detection (STD) have been actively researched in order to realize efficient searching of vast quantities of audiovisual data [5,6,7]. STD is the task of finding matching sections in speech data using a query consisting of one or more words. Query terms are often OOV words that are not contained in an automatic speech recognition (ASR) dictionary, such as technical terms, geographical names, and personal names. Because OOV query terms must be retrievable via STD systems, subword recognition using monophones, triphones, and so on is performed in advance for the speech data, and matching between a subword sequence of a query and subword sequences of the speech data is performed using a continuous dynamic programming (CDP) algorithm that continuously applies a dynamic time warping algorithm at a subword level [8].

In previous research [9], since a high-ranking candidate in an STD result showed a high precision rate and the query frequently appeared in the specific document (or presentation speech), the document that included the high-ranking candidate was assumed to contain multiple queries. As a result, retrieval accuracy improved. Matching distances to all candidates in the document that included the high-ranking candidate were used to advantage.

Speech data are generally divided into so-called “spoken documents” by topics, dialogue, sessions, lectures, and so on, where a “spoken document” refers to a presentation speech in the NTCIR evaluation set. For example, if a query is “New York”, which is spoken in such a document, words related to “New York,” such as "The Statue of Liberty", "Manhattan," and the like, can be assumed to be spoken in the document. In this paper, words co-occurring with the query in the specific document are referred to as “words related to the query,” and we can assume that the words related to the query and the query itself occur multiple times in the document. After extracting the words related to the query, distances to all the candidates in the document, including the related words, are used advantageously. Word2vec is used to identify the words related to the query in this study. Word2vec is a method of finding a distributed representation of words, wherein the degree of similarity between any two words can be calculated by comparing two word vectors. In this method, each word in a word recognition result of spoken documents is converted into a word vector, which is compared with a word vector of the query. Words related to the query are acquired by calculating the degree of similarity between the words. However, a word vector of an OOV query cannot be obtained, since OOV queries do not appear in the word-recognition results. To process such OOV queries, we utilize titles and snippets (hereinafter referred to as “Web texts”) using a Web search because a long retrieval time is required for extracting content from Web page. Web texts are thought to contain the meaning, topic, and so forth of search words. We use words in Web texts and a word recognition result of spoken documents for Word2vec, which can learn the word meaning of an OOV query and obtain its word vector. Thus, words related to the query can be acquired by calculating the degree of similarity between the word vector of the query and that of each word in spoken documents. Spoken documents including related words are then extracted, and matching distances to all candidates in those spoken documents are used advantageously according to the frequency of occurrence of the related words. Results of experiments are used to evaluate the proposed method using open test collections distributed from the National Institute of Informatics (NII) for STD evaluation in the NTCIR workshops.

Section 2 describes the proposed method, followed by the document priority method [9], in detail. Section 3 describes the evaluation experiment of the proposed method, and compares the results with previous research. Section 4 presents our conclusion.

II. PROPOSED METHOD

A. Related Work

We briefly described a related work on a document priority method [9], and mentioned the factor of a high-ranking candidate. As described in the Introduction, it is assumed that the document that includes the high-ranking candidate contains multiple queries. Let the spoken document be Ω_i(A,B,C, ...). For example, if the highest-ranking candidate is included in spoken document A, there is a high possibility that spoken document A contains multiple queries. Therefore, rescoring is performed on the matching distances to the lower-ranking candidates in spoken document A using the distance to the high-ranking candidate in accordance with the Equation (1) below. α represents a weighting coefficient $(0 \leq \alpha \leq 1)$. Let $D(\Omega_j, k)$ be the matching distance when the *j*-th candidate of spoken document Ω is ranked as k-th in Ω. The rescored distance $D'(\Omega_j, k)$ is obtained by linearly combining the original matching distance $D(\Omega_j, k)$ with the average distance of the top T ranked candidates in spoken document Ω.

$$D'(\Omega_j, k) = \alpha D(\Omega_j, k) + (1 - \alpha)\frac{1}{T}\sum_{t=1}^{T} D(\Omega_j, t) \qquad (1)$$

B. Proposed Rescoring Method Using Web Search and Word Vectors

Next, the proposed rescoring method is described. The proposed method is composed of six procedures, as described in Fig. 1.

Step 1: Pre-processing of spoken documents

Spoken documents are recognized by using a word-based ASR in advance. Given a query, the query and spoken documents are compared using CDP that applied DTW continuously at a subword-level in order to process the case of an OOV query, and candidates are outputted in order of matching distances. This retrieval result denotes baseline. The proposed method is then applied to this result.

Step2: Acquisition of Web texts

To deal with an OOV query by using Word2vec, a Web search is performed using a query, and the Web texts that include the query are extracted from the top S search results. Word2vec are trained with these Web texts (Step 3). If a query contains an OOV word, a word vector of the OOV word cannot be obtained, because OOV queries do not appear in recognized texts of the spoken documents described above. To reduce extraction time for Web texts, only the title and snippet are used in the Web search results, and the value of S is limited to 100 in this paper.

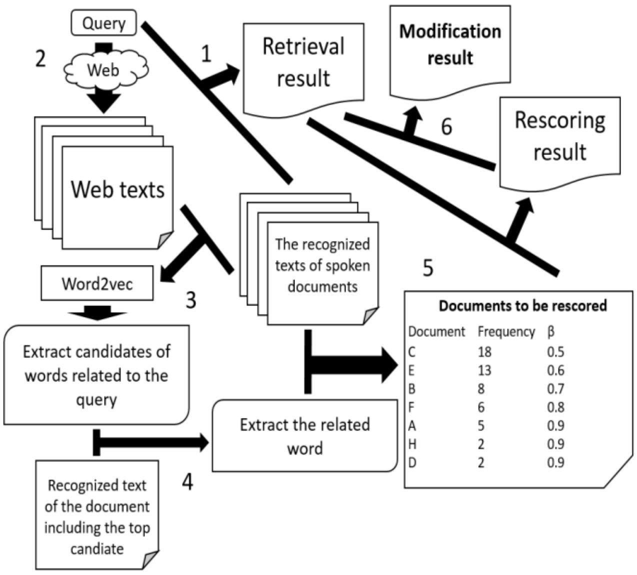

Fig. 1 Proposed Method

Step 3: Word2vec

The recognized text of the spoken documents and the Web texts are used for training Word2vec, and word vectors for all words that have appeared are obtained. By learning the Web texts, semantic information for the query can be complemented, and the degrees of relevance (cosine similarity) between the words can be computed more accurately. The CBoW (continuous bag of words) model was used for fast training in this paper.

Step 4: Extract related words

Word vectors of each word in the spoken documents and the query are computed in Step 3, and all similarities among them are also computed to obtain multiple related words. Related words are limited to only nouns in this paper, because nouns are thought to be more suitable for this purpose. The top $N(= 100, 200, \ldots, 500)$ nouns with high degrees of similarity are extracted, and set as candidates for words related to the query. As shown in [9], the top candidates in the retrieval results show high precision rates, and the spoken document that includes the top candidates has a high possibility of including the query. A word related to the query is therefore likely to appear in the spoken document that includes the top candidates. For example, if a query is “speech recognition” that is spoken in document A, it can be assumed that words related to “speech recognition,” such as "decoder" and "acoustic model," are also spoken in document A. Such related words are considered to characterize words around the query. We use tf-idf (term frequency - inverse document frequency) for determining the rank of N candidates for the related word. A single word that shows the highest tf-idf is treated as the word related to the query in this paper. The case of related words with more than one word is a topic for future research.

Step 5: Rescoring for spoken documents, including related word

In spoken documents that include the related word, the frequency of the related word corresponds to the likelihood that the query is included in the spoken documents. Multiple spoken documents that include the related word are identified by searching the recognition results of spoken documents for the related word. The amount of rescoring is determined by the frequency of the related word in each identified spoken document. Rescoring is performed for all candidates in the identified spoken documents so that a matching distance is reduced by Equation (2). When $D(\Omega_j, k)$ represents the matching distance to the j-th candidate at the k-th rank in spoken document Ω, $\mathrm{new}D(\Omega_j, k)$ represents the rescored matching distance. The coefficient β is a rescoring parameter that minimizes the original matching distance, and is determined by stepwise variation from 0.5 to 0.9, according to the frequency of the related word in this paper.

$$\mathrm{new}D(\Omega_j, k) = \beta \times D(\Omega_j, k) \qquad (2)$$

Although the rank of each candidate in spoken document Ω, which includes the related word, is the same, the distances in spoken document Ω are reduced, and the distances in other spoken documents are not changed. As the result, the ranks of the candidates in spoken document Ω become higher.

Step 6: Modification

Although a spoken document that includes a related word is extracted in Step 5, a spoken document that does not include a query might also be extracted. In such a case, rescoring has not been performed correctly, and retrieval accuracy is degraded. Therefore, the distance obtained by Equation (2) is modified by combination with the original matching distance. Modification is performed by Equation (3). γ represents the weighting coefficient $(0 \leq \gamma \leq 1)$ in the modification. The matching distance $newD'(\Omega_j, k)$ after modification is obtained by linearly combining the matching distance $newD(\Omega_j, k)$ that had been obtained by Equation (2) with the original matching distance $\mathrm{D}(\Omega_j, k)$.

$$\mathrm{new}D'(\Omega_j, k) = \gamma \times newD(\Omega_j, k) + (1 - \gamma) \times D(\Omega_j, k) \qquad (3)$$

III. EVALUATION EXPERIMENTS

A. Experimental Conditions

For acoustic and language model training, we used 1,255 presentation speeches (totaling approximately 287 h, with approximately 14 min per speech) included in the Corpus of Spontaneous Japanese (CSJ) [10]. We used a DNN-HMM hybrid speech recognizer, a triphone acoustic model composed of a left-to-right HMM composed of three states, and tied-state triphone models with 3238 states and 32 mixtures per state. Input feature vectors were extracted under the conditions shown in Table I. Five frame-feature vectors were added before and after the current frame as feature vectors for the DNN. The DNN was trained using a 1,320-dimensional feature vector under the conditions shown in Table II. Alignments between speech signals and each state were obtained from the results of forced alignment. Word trigrams were used for language models. We used the Kaldi toolkit [11] for word based speech recognition. All spoken documents were transformed into triphone sequences after word recognition using a DNN–HMM-based Kaldi toolkit. The triphone sequences were disassembled into state sequences, and an acoustic distance [8] between the states was used as a local distance. After converting the query to a state sequence via a triphone sequence, continuous DP (CDP: Continuous Dynamic Programming) was applied to perform matching between a state sequence of the query and state sequences of the spoken documents.

TABLE I Conditions for Feature Extraction

Feature parameter	120 dimensions (dim) FBANK (40 dim) + Delta-FBANK (40 dim) + Delta-Delta-FBANK (40 dim)
Window length	25 ms
Frame shift	10 ms

TABLE II Conditions for DNN

Number of nodes		Input layer: 1,320 Hidden layer: 2,048 Output layer: 3,238
Number of hidden layers		5 layers
RBM	Learning Rate	0.004
	Momentum	0.9
	Mini-batch Size	256
	Epoch	10
DNN	Learning Rate	0.007 (When recognition rate is lower than in previous epoch, it is reduced by half)
	Mini-batch Size	256
	Epoch	30

TABLE III Two Open Test Collections

	NTCIR-10	NTCIR-12
Spoken documents	104 presentations, 29 h, 40,746 utterances	98 presentations, 27.5 h, 37,782 utterances
Query sets	Formal run: 100	Formal run: 100

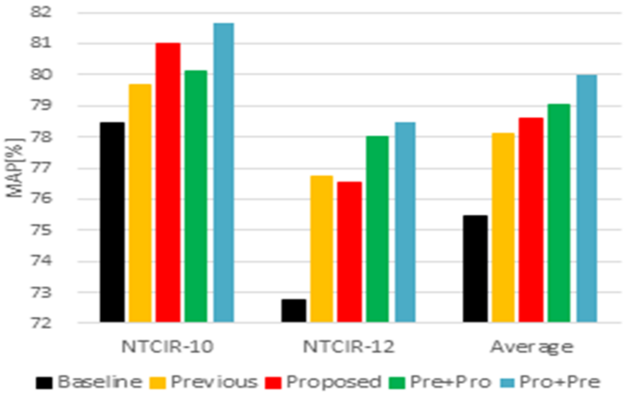

Fig. 2 Retrieval Accuracy of Each Method

B. Test Collections

To evaluate the retrieval accuracy of the proposed method, we used the two open test collections that were used in the NTCIR-10 workshop and the NTCIR-12 workshop [3,4]. As shown in Table III , the test collection of NTCIR-10 contained 29 h of presentation speeches and 100 query terms in a formal run set. The test collection of NTCIR-12 contained 27.5 h of presentation speeches and 113 query terms in a formal run set (single-term). We used mean average precision as a measure of retrieval accuracy. Cross-validation between two test sets was performed in order to determine the parameters γ and N. As described in Section 2.2, the parameter β was tested from 0.5 to 0.9 for each step of 0.1, and S was set to 100.

C. Results

We show the results when applying the proposed method described in Section II.B. In comparison with the baseline, the average improvement of the retrieval accuracy was 3.7 points in MAP for NTCIR-10 and NTCIR-12. When omitting step 6, the average improvement was only 1.6 points. The retrieval accuracy was highest at N = 300. In this case, the average processing time for a query in NTCIR-10 and NTCIR-12 was 2.8 second, which is thought to be a practical processing time.

We compared the proposed method with the previous method described in Section II.A and the method combining both methods. The results are shown in Fig. 2. Pre+Pro in the figure denotes the result from combining the two methods, in which the previous method was applied followed by the proposed method. Mean average precision (MAP) and the parameters were obtained by cross-validation between NTCIR-10 and NTCIR-12 test sets. Parameters that showed the best MAP for NTCIR-10 were used to obtain MAP for NTCIR-12, and vice versa. The baseline denotes the results without applying the rescoring methods. MAP was improved for both test sets by the proposed method and the previous method, compared with baseline accuracy. The proposed method improved retrieval accuracy by an average of 0.5 points in MAP in comparison with the previous method.

When combining the two methods in comparison with the baseline, the average improvement in MAP for the two test sets was 3.6 points in previous+proposed, and 4.6 points in proposed+previous, which achieved the highest retrieval accuracy by using the proposed method followed by the previous method. The proposed method using the related words had an additional effect on the result of the previous method, which prioritized high-ranked candidates, because both methods utilized different characteristics of language information.

The parameters with the highest retrieval accuracy in previous+proposed were γ = 0.1, N = 400 in NTCIR-10, and γ = 0.5, N = 300 in NTCIR-12, respectively. There was a difference in parameter γ. Although there was almost no difference in N, the difference in γ was substantial. If γ could be set more appropriately, retrieval accuracy might potentially be improved more.

When analyzing each query, average precision (AP) of a query for the term, "articulation" (OOV), improved by 21.4 points (67.7%→89.1%). On the other hand, a word related to “articulation” was "pronunciation" (“*hatsuon*” in Japanese); multiple occurrences of the word "pronunciation" appeared in the spoken document that included the term "articulation." The reason is that there were many articles that explained the meaning of the word “articulation” in the Web texts. In this case, the related word worked well. The AP of a query for another term, "API," decreased by 16.6 points (36.1% → 19.5%), and a word related to “API” was "specification" (“*siyou*” in Japanese). Although there were many articles written about API specifications in the Web texts, the term "specification" did not appear at all in the spoken document that included the term "API." We could not confirm which queries tended to improve retrieval accuracy at this time, and this remains a topic for future research.

IV. CONCLUSION

We have proposed a rescoring method that identifies a word related to a query using Web search and a word vector for STD. Distances to all candidates in the spoken documents that include the related word are used advantageously in the proposed method. The proposed method improved retrieval accuracy by 3.2 points in MAP, thus confirming its effectiveness. The retrieval accuracy was further improved by 1.4 points in MAP by using the proposed method followed by the previous method, thereby achieving a cumulative improvement of 4.6 points in MAP. The effectiveness of combining both the proposed method and the previous method could also be confirmed.

In the future, we hope to automatically determine parameters in the proposed method, such as the rescoring coefficient β and the number of related words N. In addition, we will evaluate other methods for extracting more appropriate words related to the query using such as BERT [12], and so on.

V. ACKNOWLEDGEMENT

This work was supported by JSPS (C), KAKENHI, Grant Number 18K11358.

REFERENCES

[1] T. Mikolov, I. Sutskever, K. Ghen, G. Corrado, J.Dean : Efficient Estimation of Words and Phrases and their Compositionally, Advances in Neaural Information Processing Systems 26, pp.3111-3119, 2013.

[2] T. Mikolov, K. Ghen, G. Corrado, J.Dean : Efficient Estimation of Word Representaions in Vector Space , Processing of the International Conference on Learning Representaions (ICLR), pp.1-12, 2013.

[3] T. Akiba, H. Nishizaki, K. Aikawa, X. Hu, Y. Itoh, T. Kawahara, S. Nakagawa, H. Nanjo, Y. Yamashita, “Overview of the NTCIR-10 SpokenDoc-2 Task,” Proc. of the 10th NTCIR Conference, pp. 573-587, 2013.

[4] T. Akiba, H. Nishizaki, H. Nanjo and G.J.F. Jones : Overview of NTCIR-12 Spoken&Doc Task, NTCIR-12, pp. 167-179, 2016.

[5] C. Auzanne, JS. Garofolo, JG. Fiscus, and WM Fisher, “Automatic Language Model Adaptation for Spoken Document Retrieval,” 2000TREC-9 SDR Track, 2000.

[6] A. Fujii, and K. itou, “Evaluating Speech-Driven IR in the NTCIR-3Web Retrieval Task,” Third NTCIR Workshop, 2003.

[7] P. Motlicek, F. Valente, and PN. Garner, “English Spoken Term Detection in Multilingual Recordings", INTERSPEECH 2010, pp.206-209, 2010.

[8] K. Iwata, Y. Itoh, K. Kojima, M. Ishigame, K. Tanaka and S. Lee, “Open-Vocabulary Spoken Document Retrieval based on new subword models and subword phonetic similarity,” INTERSPEECH, 2006.

[9] K. Konno, Y. Itoh, K. Kojima, M. Ishigame, T. Tanaka and S. Lee, “High Priority in Highly Ranked Documents in Spoken Term Detection,” 4 pages, Asia-Pacific Signal and Information Processing Association APSIPA, 2013.

[10] National Institute for Japanese Language and Linguistics, Corpus of Spontaneous Japanese, http://www.ninjal.ac.jp/corpus_center/csj/

[11] D. Povey, A. Ghoshal, G.Boulianne, L. Burget, O.N. Goel, M. Hannemann, P. Motlicek, Y. Oian, P. Schwarz, J. Silovsky, G. Stemmer and K.Vesely : The Kaldi Speech Recognition Toolkit, ASRU, 2011.

[12] J. Delvin, M. Chang, K. Lee, K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, arXiv preprint arXiv:1810.04805, 2019.